**Assessment 4 Coding exercise**

MA5800 Fundamentals of Data Science

*This is to be done during the lesson in Week 12. No communication with other students is allowed for this assessment.*

It is your choice whether you use pipes or not in your code for all questions except Table 7, where you must use pipes.

Start with a new R script in RStudio.

There are 9 Tables, 8 of which you must enter your code or input. When you have finished, upload this completed word file saved as a pdf file, please) in the **Assignment area**. You may also (optionally) include your Rscript file.

**Table 1** Copy this code into your RStudio session and run it to load the file “Internet.csv” into a dataframe called data.

Table 1.

|  |
| --- |
| library(dplyr)  library(ggplot2)  data = read.csv("Internet.csv") |

This dataset gives data for internet traffic monitored from a certain site. The data includes the time the packet was sent as well as other information which includes the size of the packet in Bytes and the source location, the destination.

VARIABLES DESCRIPTIONS:

\* Timestamp: time of packet arrivals. In minutes since the last hour.

\* Source: Source of the packet or host, with a code for confidentiality reasons.

\* Destination: Destination host with code for confidentiality reasons.

\* Sourceport: source TCP port

\* Destport: Destination TCP port

\* Databytes: number of data bytes in the packet.

Q1. Produce a histogram of the Databytes variable using 20 bins. Copy your code into Table 2. Then copy your histogram into Table 3.

**Table 2** code to produce histogram with 20 bins.

|  |
| --- |
| ggplot(data, aes(databytes)) +  geom\_histogram(bins = 20) |

**Table 3.** Copy of the histogram produced by your code in table 2.

|  |
| --- |
|  |

Q2. Using a suitable dplyr function, to write code **remove** all the rows that have databytes less than or equal to 32 and save the table in a new dataframe called data2.

**Table 4.** Code for removing rows from data frame

|  |
| --- |
| data2 = filter(data,  databytes < 32) |

**Table 5.** Output of first 4 lines

|  |
| --- |
| > data2[1:4,]  X timestamp source destination databytes  1 29 0.884258 5 16 17  2 30 0.961360 5 55 19  3 45 1.180561 43 2 29  4 63 1.813008 75 5 31 |

Starting from the original dataframe, data, write dplyr code to add an additional column to the original dataframe, called kbytes which is the databytes divided by 1000. Give the code in Table 6 and the first 4 lines output in Table 7.

**Table 6.** Code for adding additional variable.

|  |
| --- |
| data = mutate(data,  kbytes =databytes/100 ) |

**Table 7.** Output from table 5 code (first 4 lines, columns 1,7,8)

|  |
| --- |
| > data[1:4,]  X timestamp source destination databytes kbytes  1 1 0.434320 9 2 64 0.64  2 2 0.448960 2 9 32 0.32  3 3 0.455793 5 4 207 2.07  4 4 0.486049 5 11 121 1.21 |

In this question you **must use pipes**. Start from the **original data frame**, data. Write code that

1. Removes the rows which has data bytes less than or equal to 32,
2. Makes an extra column Mbytes which is the databtytes/1000
3. produces a summary of the median number of databytes for **each** value of the column source.

(Hint: it should produce a table of the median for each possible value of source, e.g. source=2, source= 9, and so on.)

Save the summary in a dataframe data\_summary. Post your code in Table 7 and the **first 4 lines** of your summary dataframe in Table 8.

**Table 8.** Code using pipes for obtaining the summary. You **must** use pipes.

|  |
| --- |
| data\_summary = data %>%  filter(databytes < 32) %>%  mutate(Mbytes = databytes/1000) %>%  group\_by(source) %>%  summarise(databytes = median(databytes)) |

**Table 9.** Give the first 4 lines of output for the dataframe produced in Table 7.

|  |
| --- |
| > data\_summary[1:4,]  # A tibble: 4 x 2  source databytes  <int> <dbl>  1 2 24  2 5 29  3 8 29  4 11 30 |